UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/603,226	06/26/2000	Donald E. Steiss	TI-29338	3213

7590 03/15/2004

Rebecca Mapstone Lake
Patents Department
Texas Instruments Incorporated
P O Box 655474 M/S 3999
Dallas, TX 75265

EXAMINER	
LI, AIMEE J	
ART UNIT	**PAPER NUMBER**
2183	13

DATE MAILED: 03/15/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/603,226	STEISS ET AL.	
	Examiner	Art Unit	
	Aimee J Li	2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *27 February 2004*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-18* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☐ Claim(s) _____ is/are rejected.

7)☒ Claim(s) *2,8,12 and 18* is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-18 have been considered. Claims 1, 8, 10, and 18 have been amended as per Applicant's request.

Allowable Subject Matter

2. Claims 2, 8, 12, and 18 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

4. Claims 1, 3, 7, 10, 12, and 16 have been rejected under 35 U.S.C. 103(a) as being unpatentable over Hammond et al., U.S. Patent Number 5,774,686 (herein referred to as Hammond) in view of O'Connor, U.S. Patent Number 5,848,288 (herein referred to as O'Connor).

5. Referring to claim 1, Hammond has taught the structure of a subpipelined translation embodiment providing binary compatibility between a base architecture and migrant architecture comprising:

a. A architecture comprising a base architecture and a migrant architecture and having a base execution mode and a migrant execution mode (Hammond Abstract, 11-17; column 2, lines 20-33; column 3, lines 41-64; column 4, lines 30-

45; column 13, lines 4-26; column 13, line 58 to column 14, line 35; column 14, line 48 to column 15, line 5; Figure 1; Figure 7; and Figure 8);

b. An instruction fetch unit for simultaneously fetching from memory a group of a plurality of instructions, each such group forming a fetch packet, said instruction fetch unit assigning each fetch packet an operating mode in dependence upon the execution mode at the time the request was made to the memory for the fetch packet (Hammond Abstract, 11-17; column 2, lines 20-33; column 3, lines 41-64; column 4, lines 30-45; column 13, lines 4-26; column 13, line 58 to column 14, line 35; column 14, line 48 to column 15, line 5; Figure 1; Figure 7; and Figure 8);

c. A shared datapath by both the base and migrant architectures for parsing said base architecture mode and migrant architecture mode fetch packets into execute packets of instructions within said fetch packet that can be executed simultaneously (Hammond Abstract, 11-17; column 2, lines 20-33; column 3, lines 41-64; column 4, lines 30-45; column 13, lines 4-26; column 13, line 58 to column 14, line 35; column 14, line 48 to column 15, line 5; Figure 1; Figure 7; and Figure 8);

d. A base architecture control circuit for dispatching execute packet instructions having a base execution mode (Hammond Abstract, 11-17; column 2, lines 20-33; column 3, lines 41-64; column 4, lines 30-45; column 13, lines 4-26; column 13, line 58 to column 14, line 35; column 14, line 48 to column 15, line 5; Figure 1; Figure 7; and Figure 8);

e. A migrant architecture control circuit for dispatching execute packet instructions having a migrant execution mode (Hammond Abstract, 11-17; column 2, lines 20-33; column 3, lines 41-64; column 4, lines 30-45; column 13, lines 4-26; column 13, line 58 to column 14, line 35; column 14, line 48 to column 15, line 5; Figure 1; Figure 7; and Figure 8);

f. A base architecture decode connected to said shared datapath and said base architecture control circuit for decoding an execute packet in said base mode and generating a corresponding machine word (Hammond Abstract, 11-17; column 2, lines 20-33; column 3, lines 41-64; column 4, lines 30-45; column 13, lines 4-26; column 13, line 58 to column 14, line 35; column 14, line 48 to column 15, line 5; Figure 1; Figure 7; and Figure 8);

g. A migrant architecture decode connected to said shared datapath and said migrant architecture control circuit for decoding an execute packet in said migrant mode and generating a corresponding machine word (Hammond Abstract, 11-17; column 2, lines 20-33; column 3, lines 41-64; column 4, lines 30-45; column 13, lines 4-26; column 13, line 58 to column 14, line 35; column 14, line 48 to column 15, line 5; Figure 1; Figure 7; and Figure 8);

h. A multiplexer having at least two inputs and one machine word output wherein one input is the machine word output of said migrant architecture decode and the other input is the machine word output of said base architecture decode, said multiplexer choosing in dependence upon the operating mode of said fetch packet (Hammond Abstract, 11-17; column 2, lines 20-33; column 3, lines 41-64;

column 4, lines 30-45; column 13, lines 4-26; column 13, line 58 to column 14, line 35; column 14, line 48 to column 15, line 5; Figure 1; Figure 7; and Figure 8); and

i. Execute hardware connected to said multiplexer for executing execute packet instructions on execution units corresponding to said machine word chosen by said multiplexer (Hammond Abstract, 11-17; column 2, lines 20-33; column 3, lines 41-64; column 4, lines 30-45; column 13, lines 4-26; column 13, line 58 to column 14, line 35; column 14, line 48 to column 15, line 5; Figure 1; Figure 7; and Figure 8).

6. Hammond has not explicitly taught a VLIW architecture comprising a base architecture and a migrant architecture and simultaneously fetching from memory a group of a plurality of instructions, each such group forming a fetch packet. However, Hammond has taught their system is usable with VLIW instruction sets (Hammond column 3, lines 41-64 and Figure 1). O'Connor has taught a VLIW architecture comprising a base architecture and a migrant architecture (O'Connor column 1, line 9 to column 2, line 2) and simultaneously fetching from memory a group of a plurality of instructions, each such group forming a fetch packet (O'Connor column 1, line 9 to column 2, line 2). A person of ordinary skill in the art at the time the invention was would have recognized that using VLIW architecture improves code execution speed of the processor by executing more than one instruction simultaneously (O'Connor column 1, lines 16-17) and allowing more than one type of VLIW instruction set to run improves compatibility of the processor with other programs by expanding the number of very long instructions recognized (O'Connor column 1, lines 60-63). Therefore, it would have been

obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the VLIW architecture and simultaneous fetching of O'Connor in Hammond to improve processor speed and compatibility.

7. Referring to claims 3 and 12, Hammond has taught wherein said machine word also controls registers (Hammond Abstract, 11-17; column 2, lines 20-33; column 3, lines 41-64; column 4, lines 30-45; column 13, lines 4-26; column 13, line 58 to column 14, line 35; column 14, line 48 to column 15, line 5; Figure 1; Figure 7; and Figure 8).

8. Referring to claims 7 and 16, Hammond has taught wherein the base and migrant architecture decode units translates opcodes to the control signals required to execute the specified instructions on the execution hardware functional units (Hammond Abstract, 11-17; column 2, lines 20-33; column 3, lines 41-64; column 4, lines 30-45; column 13, lines 4-26; column 13, line 58 to column 14, line 35; column 14, line 48 to column 15, line 5; Figure 1; Figure 7; and Figure 8).

9. Referring to claim 10, Hammond has taught a method of providing binary compatibility between a base architecture and a migrant architecture comprising the steps of:

a. Simultaneously fetching from a memory a group of a plurality of instructions, each such group forming a fetch packet (Hammond Abstract, 11-17; column 2, lines 20-33; column 3, lines 41-64; column 4, lines 30-45; column 13, lines 4-26; column 13, line 58 to column 14, line 35; column 14, line 48 to column 15, line 5; Figure 1; Figure 7; and Figure 8);

b. Assigning each fetch packet an operating mode in dependence upon the execution mode at the time the request was made to the memory for the fetch packet

(Hammond Abstract, 11-17; column 2, lines 20-33; column 3, lines 41-64; column 4, lines 30-45; column 13, lines 4-26; column 13, line 58 to column 14, line 35; column 14, line 48 to column 15, line 5; Figure 1; Figure 7; and Figure 8);

c. Parsing said base architecture mode and migrant architecture mode fetch packets into execute packets of instructions within said fetch packet that can be executed simultaneously (Hammond Abstract, 11-17; column 2, lines 20-33; column 3, lines 41-64; column 4, lines 30-45; column 13, lines 4-26; column 13, line 58 to column 14, line 35; column 14, line 48 to column 15, line 5; Figure 1; Figure 7; and Figure 8);

d. Dispatching execute packet instructions having a base execution mode dispatching execution mode (Hammond Abstract, 11-17; column 2, lines 20-33; column 3, lines 41-64; column 4, lines 30-45; column 13, lines 4-26; column 13, line 58 to column 14, line 35; column 14, line 48 to column 15, line 5; Figure 1; Figure 7; and Figure 8);

e. Dispatching execute packet instructions having a migrant execution mode (Hammond Abstract, 11-17; column 2, lines 20-33; column 3, lines 41-64; column 4, lines 30-45; column 13, lines 4-26; column 13, line 58 to column 14, line 35; column 14, line 48 to column 15, line 5; Figure 1; Figure 7; and Figure 8);

f. Decoding an execute packet in said base mode and generating a corresponding machine word (Hammond Abstract, 11-17; column 2, lines 20-33; column 3, lines

41-64; column 4, lines 30-45; column 13, lines 4-26; column 13, line 58 to column 14, line 35; column 14, line 48 to column 15, line 5; Figure 1; Figure 7; and Figure 8);

g. Decoding an execute packet in said migrant mode and generating a corresponding machine word (Hammond Abstract, 11-17; column 2, lines 20-33; column 3, lines 41-64; column 4, lines 30-45; column 13, lines 4-26; column 13, line 58 to column 14, line 35; column 14, line 48 to column 15, line 5; Figure 1; Figure 7; and Figure 8);

h. Choosing one machine word output, in dependence upon the operating mode of said fetch packet, between the machine word decoded in said migrant mode and the machine word decoded in said base mode (Hammond Abstract, 11-17; column 2, lines 20-33; column 3, lines 41-64; column 4, lines 30-45; column 13, lines 4-26; column 13, line 58 to column 14, line 35; column 14, line 48 to column 15, line 5; Figure 1; Figure 7; and Figure 8);

i. Controlling machine word the execution hardware units with said chosen machine word (Hammond Abstract, 11-17; column 2, lines 20-33; column 3, lines 41-64; column 4, lines 30-45; column 13, lines 4-26; column 13, line 58 to column 14, line 35; column 14, line 48 to column 15, line 5; Figure 1; Figure 7; and Figure 8).

10. Hammond has not explicitly taught a VLIW architecture and simultaneously fetching from a memory a group of a plurality of instructions, each such group forming a fetch packet. However, Hammond has taught their system is usable with VLIW instruction sets (Hammond column 3, lines 41-64 and Figure 1). O'Connor has taught a VLIW architecture comprising a

base architecture and a migrant architecture (O'Connor column 1, line 9 to column 2, line 2) and simultaneously fetching from memory a group of a plurality of instructions, each such group forming a fetch packet (O'Connor column 1, line 9 to column 2, line 2). A person of ordinary skill in the art at the time the invention was would have recognized that using VLIW architecture improves code execution speed of the processor by executing more than one instruction simultaneously (O'Connor column 1, lines 16-17) and allowing more than one type of VLIW instruction set to run improves compatibility of the processor with other programs by expanding the number of very long instructions recognized (O'Connor column 1, lines 60-63). Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the VLIW architecture and simultaneous fetching of O'Connor in Hammond to improve processor speed and compatibility.

11. Claims 4-6, 9, 13-15, and 17 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hammond in view of O'Connor as applied to claims 1 and 10 above, and further in view of Nishioka et al., U.S. Patent Number 6,401,190 (herein referred to Nishioka).

12. Referring to claims 4-9, Hammond has taught a VLIW architecture further comprising wherein said machine word controls the various types of execution hardware that evaluate functions on the operands to produce the results of said hardware execution units subsequent to said machine word controlling said local register files (Applicant's claim 6) (Hammond Abstract, 11-17; column 2, lines 20-33; column 3, lines 41-64; column 4, lines 30-45; column 13, lines 4-26; column 13, line 58 to column 14, line 35; column 14, line 48 to column 15, line 5; Figure 1; Figure 7; and Figure 8). Hammond has not explicitly taught a VLIW architecture further comprising:

a. Wherein said machine word controls a global register file, which supplies operands to all hardware execution units and accepts results of all hardware execution units (Applicant's claim 4);

b. Wherein said machine word controls local register files that supply operands to either local execution hardware functional units or neighbor hardware execution functional units subsequent to said machine word controlling said global register file (Applicant's claim 5);

c. Wherein said VLIW architecture is a Digital signal Processor (DSP) (Applicant's claim 9).

13. However, Hammond has taught that the VLIW instruction set architecture is one of many possible instructions sets that can be used in the device (Hammond column 3, lines 41-64 and Figure 1), but not the explicit details of the VLIW instruction set architecture. Nishioka has taught explicitly a VLIW architecture (Nishioka column1, line 41 to column 2, line 4) further comprising:

a. Wherein said machine word controls a global register file, which supplies operands to all hardware execution units and accepts results of all hardware execution units (Applicant's claim 4) (Nishioka column 4, lines 7-62);

b. Wherein said machine word controls local register files that supply operands to either local execution hardware functional units or neighbor hardware execution functional units subsequent to said machine word controlling said global register file (Applicant's claim 5) (Nishioka column 4, lines 7-62);

c. Wherein said VLIW architecture is a Digital signal Processor (DSP) (Applicant's claim 9) (Nishioka column 1, lines 14-41).

14. A person of ordinary skill in the art at the time of applicant's invention would have recognized that a VLIW architecture simplifies hardware and how the hardware is controlled, since no decoder is needed to translate the instruction from higher level instructions to machine level instructions is required (Nishioka column 1, lines 50-57). Hardware simplification would have motivated one of ordinary skill in the art to incorporate VLIW architecture. Therefore, it would have been obvious to a person of ordinary skill in the art at the time this invention was made to incorporate a VLIW architecture as taught by Nishioka in the device of Hammond to simplify hardware.

15. Referring to claims 11, 13-15, and 17, Hammond has taught a VLIW architecture method further comprising controlling the various types of execution hardware that evaluate functions on the operands to produce the results of said hardware execution units subsequent to controlling said local register files (Applicant's claim 15) (Hammond Abstract, 11-17; column 2, lines 20-33; column 3, lines 41-64; column 4, lines 30-45; column 13, lines 4-26; column 13, line 58 to column 14, line 35; column 14, line 48 to column 15, line 5; Figure 1; Figure 7; and Figure 8).

16. Hammond has not explicitly taught a VLIW architecture method further comprising:

a. Choosing between the output of said migrant architecture decode and the output of said base architecture decode input and a no operation instruction (Applicant's claim 11);

b. Controlling a global register file with said machine word, which supplies operands to all hardware execution units and accepts results of all hardware execution units (Applicant's claim 13);

c. Controlling local register files that supply operands to either local execution hardware functional units or neighbor hardware execution functional units subsequent to said controlling said global register file (Applicant's claim 14);

d. Wherein said VLIW architecture is a Digital Signal Processor (DSP) (Applicant's claim 17); and

17. However, Hammond has taught that the VLIW instruction set architecture is one of many possible instructions sets that can be used in the device (Hammond column 3, lines 41-64 and Figure 1), but not the specific details of the VLIW architecture. Nishioka has taught explicitly a VLIW architecture method (Nishioka column1, line 41 to column 2, line 4) further comprising:

a. Choosing between the output of said migrant architecture decode and the output of said base architecture decode input and a no operation instruction (Applicant's claim 11) (Nishioka column 7, lines 31-59; column 8, line 47 to column 9, line 20; column 14, lines 30-37; Figure 5, and Figure 6);

b. Controlling a global register file with said machine word, which supplies operands to all hardware execution units and accepts results of all hardware execution units (Applicant's claim 13) (Nishioka column 4, lines 7-62);

c. Controlling local register files that supply operands to either local execution hardware functional units or neighbor hardware execution functional units

subsequent to said controlling said global register file (Applicant's claim 14) (Nishioka column 4, lines 7-62);

d. Wherein said VLIW architecture is a Digital Signal Processor (DSP) (Applicant's claim 17) (Nishioka column 1, lines 14-41); and

18. A person of ordinary skill in the art at the time of applicant's invention would have recognized that a VLIW architecture simplifies hardware and how the hardware is controlled, since no decoder is needed to translate the instruction from higher level instructions to machine level instructions is required (Nishioka column 1, lines 50-57). Hardware simplification would have motivated one of ordinary skill in the art to incorporate VLIW architecture. Therefore, it would have been obvious to a person of ordinary skill in the art at the time this invention was made to incorporate a VLIW architecture as taught by Nishioka in the device of Hammond to simplify hardware.

Response to Arguments

19. Applicant's arguments with respect to claims 1-18 have been considered but are moot in view of the new ground(s) of rejection.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Aimee J Li whose telephone number is (703) 305-7596. The examiner can normally be reached on M-T 7:30am-5:00pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (703) 305-9712. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

AJL
Aimee J. Li
March 10, 2004

EDDIE CHAN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100